# Requirements – Student #5

Please, fill in the following form, make sure that you have ticked the requirements that you consider fulfilled, save this document, **and attach it in its original format (.docx)** to every deliverable. Regarding your ID, please keep only four random digits and mask the others using an asterisk. **Please, note that this document must be edited with the desktop version of Word since the web version does not properly support forms.** Attaching this document entails that you are the authors of the work delivered, you have not cheated in any way, and you have read and understood the information delivered regarding the subject, with a special emphasis on the methodological guidelines and how your work is going to be graded. Make sure that your project works well with the latest version of the development framework.

|  |
| --- |
| **Group:** C1.056 |
| **Repository:** https://github.com/AngSanRui/Acme-ANS-C2 |
| **Student #2**  **ID Number:** \*\*63\*\*98  **UVUS:**  CRP8717  **Name:**  López Expósito, Rubén  **Roles:**  developer, tester |
| **Date:** Sevilla July 04 , 2025 |

# MANDATORY Deliverable D01: introduction

## Information requirements

Intentionally blank.

## Functional requirements

1. Modify the anonymous menu so that it shows an option that takes the browser to the home page of your favourite web site. The title must read as follows: “〈id-number〉: 〈surname〉, 〈name〉”, where “〈id-number〉” denotes your DNI, NIE, or passport number, “〈surname〉” denotes your surname/s, and “〈name〉” denotes your name/s.

x

## Non-functional requirements

Intentionally blank.

## Testing requirements

Intentionally blank.

## Managerial requirements

1. Provide a link to your planning dashboard in GitHub to review the tasks, their current status, and your schedule.

x El enlace no funciona.

RL: Se ha puesto en público el tablero para que funcione.

# MANDATORY Deliverable D02: data models

## Information requirements

1. The **technicians** care of aircraft maintenance by conducting regular inspections, performing repairs, and carrying out other maintenance tasks. The system must store the following data about them: a **license number** (unique, pattern "^[A-Z]{2-3}\d{6}$"), a **phone number** (pattern "^\+?\d{6,15}$"), their **specialisation** (up to 50 characters), whether they have passed their **annual health test** or not, and their **years of experience**. Optionally, the system may store their **certifications** (up to 255 characters).

x

1. **Maintenance records** are comprehensive records of activities performed on a given **aircraft** throughout its operational life. The system must store the following data about them: the **moment** when a maintenance takes place, its **status** ("PENDING", "IN PROGRESS", "COMPLETED"), the next **inspection due date**, an **estimated cost**, and some optional **notes** (up to 255 characters).

x

1. **Maintenance records** rely on **tasks**. A task is a specific predefined operational duty carried out by a **technician** on **aircrafts**. The system must store the following data about tasks: their **type** ("MAINTENANCE", "INSPECTION", "REPAIR", "SYSTEM CHECK"), a **description** (limited to 255 characters), a **priority** (ranging from 0 to 10), and an **estimated duration** (in hours).

x

## Functional requirements

Intentionally blank.

## Non-functional requirements

Intentionally blank.

## Testing requirements

1. Produce assorted sample data to test your application informally. The data must include two **technician** accounts with credentials “**technician1**/ **technician1**” and “**technician2**/ **technician2**”. Create an additional technician account with credentials “**technician3/ technician3”** that represents a technician with no data, but his or her profile.

x

## Managerial requirements

1. Provide a link to your planning dashboard in GitHub to review the tasks, their current status, and your schedule.

x Enlace no funciona.

RL: Se ha puesto en público el tablero para que funcione.

# MANDATORY Deliverable D03: implementing features

## Information requirements

Intentionally blank.

## Functional requirements

1. Operations by **technicians** on **maintenance records**:

* List their maintenance records.
* Show the details of the maintenance records, including the tasks involved.
* Create, update, and publish maintenance records. Please, note that to publish a maintenance record, it cannot have any unpublished tasks and should have at least one published task.

x Creo un registro de mantenimiento con los siguientes datos:

A screenshot of a computer

AI-generated content may be incorrect.

Accedo a la lista de tareas y compruebo que las dos que le he añadido están publicadas:

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

Al pulsar sobre el botón para publicar el registro de mantenimiento ocurre lo siguiente:

A screenshot of a computer

AI-generated content may be incorrect.Cambio el estado a COMPLETED de la siguiente forma:

A screenshot of a computer

AI-generated content may be incorrect.

Pulso de nuevo el botón para publicar y el resultado es el que se ve. Me indica que no se puede publicar a pesar de tener dos tareas publicadas.

RL: Al ver las tareas no se accede como tal a la propia tarea, sino que se accede a un objeto de tipo Involves, que representa la relación entre una tarea y un registro de mantenimiento, y el cual no debe permitir editar la tarea, ya que únicamente muestra detalles de la tarea relacionada con un registro de mantenimiento. El error por tanto proviene de que no se llegan a publicar las tareas, ya que el hecho de que aparezcan como read-only al entrar en el objeto Involves que relaciona esa tarea con un registro no significa que esté publicada, simplemente que no se puede editar esta tarea desde la relación. Por lo tanto, al no estar publicadas es normal que aparezca un mensaje de error que no te permita publicar un registro de mantenimiento, ya que para ello se tienen que publicar las tareas desde la propia lista de tareas, para así poder llegar a publicar el registro de mantenimiento.

RC: Ese objeto al que Ud. hace referencia la forma en que se implementan habitualmente objetos conglomerados de tipo agregado débil, que es justamente su caso: un registro de mantenimiento está formado por una o varias tareas, cada una de las cuales puede formar parte de cero, uno o varios registros de mantenimiento.

Su cliente y los usuarios de su cliente entienden de registros de mantenimiento y de tareas, no de los objetos intermedios que ud. tiene que introducir para implementar los requisitos. Dichos objetos son parte de su implementación de los requisitos, no de los requisitos con los que trabaja su cliente o sus usuarios.

Lo que vemos en la interfaz de usuario es un registro de mantenimiento en estado COMPLETED que NO está publicado que tiene dos tareas SÍ publicadas. Por lo tanto debería ser posible publicarlo y el mensaje de error indica que no es posible porque tiene tareas sin publicar.

Tendrá Ud. que arreglar lo que comenta de forma que el usuario pueda trabajar con sus registros de mantenimiento y tareas sin saber nada sobre cómo ha implementado Ud. el requisito.

RL: He solucionado este problema modificando la vista del formulario de las tareas desde el objeto involves para que ahora se puedan realizar las tareas de actualización, eliminación y publicación de tareas desde la propia relación, para que así cada usuario solo tenga que saber de tareas y registros de mantenimientos, y por tanto no se vea afectado porque haya una clase involves en medio que los relacione.

Para conseguir esto he modificado el frontend, para que aparezcan los mismos botones de las acciones que se pueden realizar sobre una tarea, así como el backend, para que funcionen estas acciones realizándolas desde la vista de show de un objeto involves.

El resultado final de este cambio con el que se comprueba que este requisito se ha arreglado se puede observar a través de las siguientes capturas:

Captura de pantalla de un celular

El contenido generado por IA puede ser incorrecto.

Esta es la pantalla donde se muestra una tarea dentro de una relación entre una tarea y un registro de mantenimiento, pudiéndose observar que están disponibles las funcionalidades de actualizar, eliminar y publicar tarea, y que la propia tarea no está publicada.

Interfaz de usuario gráfica, Texto, Aplicación

El contenido generado por IA puede ser incorrecto.

Aquí se ve que no se puede publicar el registro de mantenimiento ya que la tarea que tiene no está publicada.

Captura de pantalla de un celular

El contenido generado por IA puede ser incorrecto.

Se ha publicado la tarea desde la propia pantalla de la relación tarea-registro de mantenimiento.

Captura de pantalla de un celular

El contenido generado por IA puede ser incorrecto.

Finalmente se ha podido publicar el registro de mantenimiento gracias a que ahora se puede realizar la gestión de las tareas desde la propia relación entre registros de mantenimientos y tareas, sin necesidad de tener en cuenta que el objeto que los relaciona existe.

1. Operations by **technicians** on **tasks**:

* List their tasks and show their details.
* Create, update, delete, and publish tasks. Note that published tasks cannot be updated or deleted.

x Como manager3, publico la siguiente tarea:

A screenshot of a computer

AI-generated content may be incorrect.

Por lo tanto, debe formar parte ya del catálogo de tareas disponible. Entro en el sistema como technician1 y al intentar verla ocurre lo siguiente:

A screenshot of a computer error message

AI-generated content may be incorrect.

Esto es un contrasentido, dado que cuando se va accede a la ficha para añadir una nueva tarea a un registro de mantenimiento, sí que sale:

A screenshot of a computer

AI-generated content may be incorrect.

¿Qué sentido tiene que no pueda añadir una tarea cuyos detalles no puedo ver?

RL: Este error en el requisito se ha solucionado consiguiendo que aparezcan las tareas publicadas por un technician en la lista de tareas de cualquier technician. Esto se puede comprobar con las siguientes capturas:

Interfaz de usuario gráfica, Texto

El contenido generado por IA puede ser incorrecto.

Esta es una tarea del technician 3 publicada.

Captura de pantalla de un celular

El contenido generado por IA puede ser incorrecto.

Esta es la misma tarea publicada del technician 3 pero viéndola logueado como technician 1, lo cual se puede comprobar en la parte inferior de la pantalla, en el apartado de “Principal”.

## Non-functional requirements

Intentionally blank.

## Testing requirements

Intentionally blank.

## Managerial requirements

1. Provide a link to your planning dashboard in GitHub to review the tasks, their current status, and your schedule.

x El enlace proporcionado no funciona.

RL: Se ha puesto en público el tablero para que funcione.

# MANDATORY Deliverable D04: formal testing

## Information requirements

1. Create appropriate indices for your entities, if required.

x Los índices no son adecuados. Por ejemplo, el estudiante introduce índices redundantes como en:

A screenshot of a computer code

AI-generated content may be incorrect.

A screenshot of a computer program

AI-generated content may be incorrect.

RL: El error comentado sobre este requisito se ha solucionado de la siguiente manera.

Por una parte se han quitado los índices que incluían id de algún objeto, ya que son redundantes.

Por otro lado, se han añadido índices de atributos que no se habían tenido en cuenta anteriormente.

Los índices añadidos se pueden observar en las siguientes capturas:

Interfaz de usuario gráfica, Texto

El contenido generado por IA puede ser incorrecto.

Interfaz de usuario gráfica, Texto

El contenido generado por IA puede ser incorrecto.

## Functional requirements

Intentionally blank.

## Non-functional requirements

Intentionally blank.

## Testing requirements

1. Produce a test suite for Requirements #8 and #9.

x Al ejecutar la suite de pruebas proporcionada se produce el siguiente fallo en todos los casos de prueba:



RL: Este fallo era provocado por un problema en un validador de otro estudiante, el cual ha podido solucionar el problema, por lo que ya no aparece dicho fallo en ningún caso de prueba al reproducir la suite de pruebas.

Por otro lado, se han repetido los casos de prueba para algunas funcionalidades que habían dejado de funcionar, además de que al repetirse se han podido probar las funcionalidades con los cambios implementados en esta entrega.

## Managerial requirements

1. Provide a link to your planning dashboard in GitHub to review the tasks, their current status, and your schedule.

x El enlace no funciona.

RL: Se ha puesto en público el tablero para que funcione.

1. Produce a testing report.

x Dados los problemas identificados anteriormente, queda claro que el informe no corresponde al proyecto entregado para evaluación.

RL: Se ha vuelto ha realizar el performance testing sobre los nuevos casos de prueba, llegando a conclusiones diferentes y más concretas que en la anterior entrega. Todo esto se ha reflejado en el nuevo testing report.

# SUPPLEMENTARY I Deliverable D01: introduction

## Information requirements

Intentionally blank.

## Functional requirements

Intentionally blank.

## Non-functional requirements

Intentionally blank.

## Testing requirements

Intentionally blank.

## Managerial requirements

Intentionally blank.

# SUPPLEMENTARY I Deliverable D02: data models

## Information requirements

1. The system must handle **technician dashboards** with the following **indicators**:

* The number of maintenance records grouped by their status.
* The maintenance record with the nearest inspection due date, provided that he or she is involved in any tasks that need to be performed as part of that maintenance.
* The top five aircrafts with higher number of tasks in their maintenance records.
* The average, minimum, maximum, and standard deviation of the estimated cost of their maintenance records in the last year.
* The average, minimum, maximum, and standard deviation of the estimated duration of the tasks in which he or she is involved.

## Functional requirements

Intentionally blank.

## Non-functional requirements

Intentionally blank.

## Testing requirements

Intentionally blank.

## Managerial requirements

1. Produce a UML domain model regarding the information requirements in your project.

# SUPPLEMENTARY I Deliverable D03: implementing features

## Information requirements

Intentionally blank.

## Functional requirements

1. Operations by **anonymous principals** on **user accounts**:

* Sign up to the system and become a technician.

1. Operations by **technicians** on **user accounts**:

* Update their profiles.

1. Operations by **administrators** on **maintenance records**:

* List the maintenance records in the system that are published.
* Show the details of the maintenance records (including the tasks involved).

1. Operations by **technicians** on **dashboards**:

* Show their dashboards.

## Non-functional requirements

Intentionally blank.

## Testing requirements

Intentionally blank.

## Managerial requirements

1. Provide a link to a video in which you informally test requirement #8 and #9. Videos should not exceed 10 minutes in length and must be stored at the USE's facilities.

# SUPPLEMENTARY I Deliverable D04: formal testing

## Information requirements

Intentionally blank.

## Functional requirements

Intentionally blank.

## Non-functional requirements

Intentionally blank.

## Testing requirements

1. Perform five mutations in your code and report on the results.

## Managerial requirements

1. Produce a lint report.

# SUPPLEMENTARY II Deliverable D01: introduction

## Information requirements

Intentionally blank.

## Functional requirements

Intentionally blank.

## Non-functional requirements

Intentionally blank.

## Testing requirements

Intentionally blank.

## Managerial requirements

1. Produce an analysis report.

1. Produce a planning and progress report.

# SUPPLEMENTARY II Deliverable D02: data models

## Information requirements

1. The system is required to have a notice board to advertise **courses** for technicians. A web service must be used to populate this entity with information about courses. Thus, the exact data to store depends on the chosen service, and it is the students' responsibility to define them accordingly. It is also the students’ responsibility to find the appropriate service; no implicit or explicit liabilities shall be covered by the University of Seville or their individual affiliates if the students hire pay-per-use services! The students are strongly advised to ensure that the service they choose is free of charge.

## Functional requirements

Intentionally blank.

## Non-functional requirements

Intentionally blank.

## Testing requirements

Intentionally blank.

## Managerial requirements

1. Produce an analysis report.

1. Produce a planning and progress report.

# SUPPLEMENTARY II Deliverable D03: implementing features

## Information requirements

Intentionally blank.

## Functional requirements

1. Operations by **technicians** on **courses**:

* List the course available and show their details.

1. Operations by **administrators** on **courses**:

* Populate the database with courses data.

## Non-functional requirements

Intentionally blank.

## Testing requirements

Intentionally blank.

## Managerial requirements

1. Produce an analysis report.

1. Produce a planning and progress report.

# SUPPLEMENTARY II Deliverable D04: formal testing

## Information requirements

Intentionally blank.

## Functional requirements

Intentionally blank.

## Non-functional requirements

Intentionally blank.

## Testing requirements

1. Produce as a complete test suite as possible for Requirement #29 ensuring that the web service is properly mocked.

## Managerial requirements

1. Produce an analysis report.

1. Produce a planning and progress report.